UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/789,523	02/26/2004	Miguel Isenberg	20910/0206101-US0	7733

62663 7590 01/10/2008

Sun Microsystems, Inc.
c/o DARBY & DARBY P.C.
P.O. BOX 770
Church Street Station
NEW YORK, NY 10008-0770

EXAMINER	
CAO, DIEM K	
ART UNIT	**PAPER NUMBER**
2194	

MAIL DATE	DELIVERY MODE
01/10/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/789,523	ISENBERG, MIGUEL	
	Examiner	Art Unit	
	Diem K. Cao	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *25 October 2007*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-44* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-44* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-44 are pending. Applicant has amended claims 1, 21, 29 and 41.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 1-13, 15, 17-33, 35 and 37-44 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bhattacharya (Design Notes on Asynchronous I/O (aio) for Linux) in view of Chase et al. (U.S. 2004/0109410 A1) further in view of Benhase et al. (U.S. 6,745,262 B1).

As to claim 1, Bhattacharya teaches in a computer system, a method for retrieving events from an event port, the method comprising:

- receiving from a computer software application, a request to retrieve a specified number of events from an event port to which completed events are posted by one or more event sources (completion queues, there are api's ... with that queue; page 8, third paragraph, and "Ability to reap many events together ... than just a single event"; page 8, last paragraph and "Ability to wait for at least a specified number ... to complete"; page 9, section 'Enable flexible grouping of operations', and completion port style; page 13, section 2.7),

- determining whether the specified number of events is available at the event port (A natural extension ... or go back to sleep; page 12, section 2.6.2 and Retrieve completion event ... to arrive; page 14, section 3.1),
- if the specified number of events is available at the event port, retrieving the specified number of events from the event port and returning the retrieved events to the requesting computer software application (A natural extension ... or go back to sleep; page 12, section 2.6.2 and When the operation complete ... ring buffer; page 18, section 4.2.2), and
- if fewer events than the specified number of events are available at the event port, placing the request in a request queue with requests to be processed at a later time (A natural extension ... or go back to sleep; page 12, section 2.6.2, and If no events are present, then wait for up to the timeout for at least one event to arrive; page 14, section 3.1 and wait queue; page 15, section 4.1.1 and page 17, section 4.2.1) and ordering the request queue based on priorities of the requests in the request queue (priority; pages 6-7, section 2.4 and page 10, section 5).

Bhattacharya does not explicitly teach ordering the request queue based on priorities of the requests in the request queue, and changing the priorities of the requests in the request queue based on the number of events available at the event port, wherein the changing is further based on a specified number of events to be retrieved as part of at least one request received in response to the number of events available at the event port.

However, Benhase teaches ordering the request queue based on priorities of the requests in the request queue (See Fig. 2 and col. 4, lines 35-55), and Chase teaches changing the

priorities of the requests in the request queue based on number of tasks of each request and the available of the resource (change a queue priority requests based on the type f request and what system resource is approaching the overload condition; page 3, paragraphs 22-23).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Benhase and Chase to the system of Bhattacharya because Benhase teaches a method and a data structure for queuing requests capable of having different priority levels (col. 1, lines 8-10), and this will improve the performance of Bhattacharya's system by avoid managing multiple priority queues by the system (col. 2, lines 38-40), and Chase teaches a system that can change the priority of requests to avoid the overload condition (page 1, paragraph7).

As to claim 2, Bhattacharya does not explicitly teach wherein ordering comprises placing requests with a higher priority ahead of requests with lower priority in the request queue. However, Benhase teaches wherein ordering comprises placing requests with a higher priority ahead of requests with lower priority in the request queue (see Fig. 2). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Benhase to the system of Bhattacharay because Benhase teaches a method and data structure for queuing requests capable of having different priority levels (col. 1, lines 8-10).

As to claim 3, Bhattacharya does not explicitly teach wherein ordering comprises placing two or more requests with a same priority in a stack, and placing the stack in the request queue based on the priority of the requests in the stack. However, Benhase teaches ordering comprises

placing two or more requests with a same priority in a stack, and placing the stack in the request queue based on the priority of the requests in the stack (abstract). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the teaching of Benhase to the system of Bhattacharya because Benhase teaches a method and a data structure for queuing requests capable of having different priority levels (col. 1, lines 8-10), and this will improve the performance of Bhattacharya's system by avoid managing multiple priority queues by the system (col. 2, lines 38-40).

As to claim 4, Bhattacharya does not teach wherein the specified number of events to be retrieved from the event port indicates a priority of the request. However, Bhattacharya teaches the priority of the request is provided by the application (pages 6-7, section 2.4) and the number of events to be retrieved is also provided by the application (page 12, section 2.6.2). It would have been obvious to one of ordinary skill in the art that the application can be implemented to have the number of events to be retrieved indicates the priority of the request.

As to claim 5, Bhattacharya does not explicitly teach wherein a priority of a request is inversely proportional to the specified number of events. See discussion of claim 4 above for the same reason.

As to claim 6, Bhattacharya teaches wherein the request queue contains requests generated by one or more computer software application threads (an application can issue a wait

on a given queue to be notified of a completion event for any request associated with that queue; page 8, third paragraph and wait queue; page 15, section 4.1.1 and page 17, section 4.2.1).

As to claim 7, Bhattacharya teaches wherein the request queue contains requests generated by one or more computer software application processes (an application can issue a wait on a given queue to be notified of a completion event for any request associated with that queue; page 8, third paragraph and wait queue; page 15, section 4.1.1 and page 17, section 4.2.1).

As to claim 8, Bhattacharya teaches wherein the number of events to be retrieved from the event port is specified by the computer software application (page 12, section 2.6.2).

As to claim 9, Bhattacharya does not explicitly teach wherein the number of events to be retrieved from the event port is specified by the computer software application based on user input. However, Bhattacharya teaches implement at-least-N semantics purely in user space (page 12, section 2.6.2). It would have been obvious to one of ordinary skill in the art at the time the invention was made to have the number of events to retrieve is specified by the user, so the user can have control over the system.

As to claim 10, Bhattacharya teaches

- if fewer events than the specified number of events are available at the event port, determining whether there are any requests in the request queue that can be satisfied by the available number of events at the event port (there are also ... for this operation; page

10, section 3, and wakeup only waiter whose "N-value" matches or exceeds the number of events available; page 12, section 2.6.2), and

- if there are requests in the request queue that can be satisfied, retrieving the specified number of events from the event port for one or more such requests and returning the retrieved events to the requesting computer software application (and then have them try to pick up its N events; page 12, section 2.6.2).

As to claim 11, Bhattacharya teaches wherein the request has an associated timeout prior to which the request must be satisfied (Wait up to the timeout for the i/o described by the specific iocb to complete; page 14, third paragraph).

As to claim 12, Bhattacharya teaches if a timeout occurs for a request while the request is in the request queue, retrieving all the available events at the event port a the time of timeout, and returning the request to the computer software application with the retrieved events (The DASF api support ... in case of a timeout; page 9, section 2 'Enable flexible grouping of operations').

As to claim 13, Bhattacharya does not explicitly teach returning an empty request to the requesting software application if the request cannot be satisfied. However, Bhattacharya teaches wait up to the timeout for the i/o to complete (page 14, third paragraph). It would have been obvious to one of ordinary skill in the art that an empty response would be returned if the i/o has not been completed by the time the timeout is up.

As to claim 15, see rejection of claim 13 above. Bhattacharya further teaches the timeout may occur before the completion event arrived to the completion queue. It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the teaching of Bhattacharya to have the error message return if the request contains an invalid event port identifier, an event or a list of events list cannot be delivered, a timeout argument is out of range, and a timeout interval expires before an expected number of events has been posted to the event port.

As to claim 17, Bhattacharya does not explicitly teach if the specified number of events is zero, identifying the number of available events at the event port, and informing the requesting computer software application of how many events are available at the event port. However, Bhattacharya teaches if no events are present, then wait for up to the timeout for at least one event to arrive (page 14, section 3.1), and also other waiter is waited on the same event (page 10, second paragraph). It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of Bhattacharya to let the application know how many events are queued in the completion queue, so they can aware as whether the queue is empty or full to have other option, such as cancel the operation (page 14, section 3.1).

As to claim 18, Bhattacharya teaches wherein the events are asynchronous events (Option to wait for notification of aio events; page 3, second paragraph).

As to claim 19, Bhattachary teaches wherein the events are transaction events (wait for all the submitted events to complete; page 13, second paragraph).

As to claim 20, Bhattachary teaches wherein the event sources include one or more of: input devices, output devices, timers, signals, file updates, applications, system libraries, and drivers (page 2, sections 1.1 and 1.2).

As to claim 21, it is the same as the method claim of claim 1 except it is a computer product claim, and is rejected under the same ground of rejection.

As to claims 22-33, see rejection of claims 2-13 above.

As to claim 35, see rejection of claim 15 above.

As to claim 37, see rejection of claim 17 above.

As to claims 38-40, see rejections of claims 18-20 above.

As to claim 41, see rejections of claims 1, 6. Bhattacharya further teaches an event queue for receiving transaction events generated by one or more event sources, the event queue being accessible through an event port (completion queues, api; page 8, third paragraph).

As to claim 42, see rejection of claim 3 above.

As to claim 44, see rejection of claim 11 above.

As to claim 43, see rejection of claim 4 above.

4. Claims 14, 16, 34 and 36 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bhattacharya (Design Notes on Asynchronous I/O (aio) for Linux) in view of Chase et al. (U.S. 2004/0109410 A1) and Benhase et al. (U.S. 6,745,262 B1) further in view of Lucovsky et a. (U.S. 6,223,207 B1).

As to claim 14, Bhattacharya does not teach wherein returning comprises returning the empty request together with an error indicating the cause for why the request cannot be satisfied. However, Lucovsky teaches for each request that related to the completion port, there is error handling if there are any type of error (col. 12, lines 44-64 and col. 13, line 46-55). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the modified the teaching of Lucovsky to the system of Bhattacharya because Lucovsky teaches a method to let the application users know the reason of the error, so the user can have option to handle the errors that raise during the execution of the system.

As to claim 16, Bhattacharya teaches wherein returning the retrieved events to the requesting computer software application comprises returning one or more of one or more

detected events (then have them try to pick up its N events; page 12, section 2.6.2). Bhattacharya does not teach one or more event source identifiers where the detected events were generated, one or more objects specific to an event source, and one or more user defined values. However, Lucovsky teaches the completion packet contains the number of bytes read/written, error indication, the context associated with the particular I/O operation, the context associated with the particular file handler (col. 13, line 64 – col. 14, line 1). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply and modify the teaching of Lucovsky to the system of Bhattacharya because Lucovsky teaches a method to return additional data to application.

As to claim 34, see rejection of claim 14 above.

As to claim 36, see rejection of claim 16 above.

Response to Arguments

5. Applicant's arguments with respect to claims 1-44 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

6. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Diem K. Cao whose telephone number is (571) 272-3760. The examiner can normally be reached on Monday - Friday, 8:30AM - 4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on (571) 272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

DC
January 2nd, 2008

WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER